A Decoder Architecture for High-Speed Free Space Laser Communications

Mike Cheng, Mike Nakashima, Jon Hamkins, Bruce Moision, and Maged Barsoum
Jet Propulsion Laboratory, California Institute of Technology
Pasadena, CA
{mkcheng, michael.a.nakashima}@jpl.nasa.gov

SPIE Photonics West
January 26, 2005
San Jose, CA

Information Processing Group

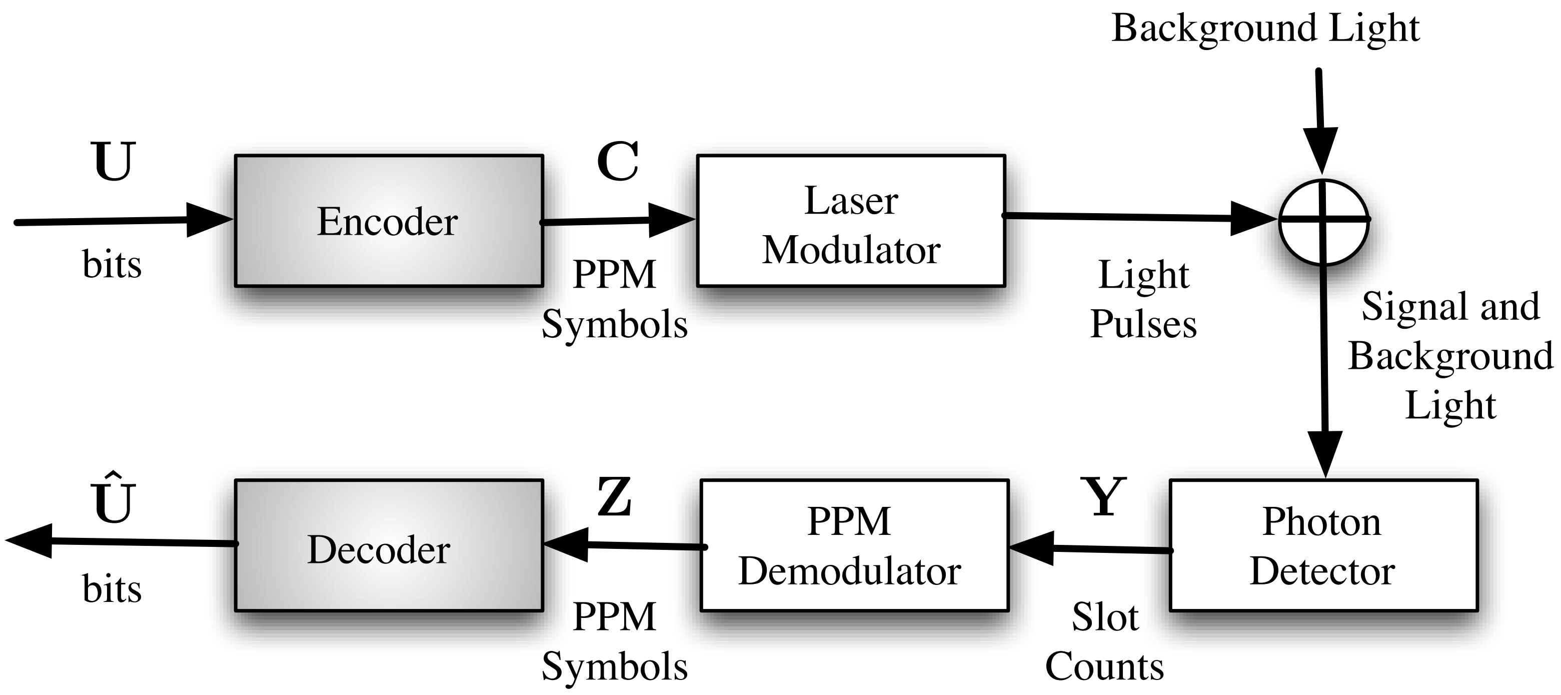
Outline

- An optical communication system
- The Serial Concatenated Pulse-Position Modulation (SCPPM) Code
- The Soft-In Soft-Out (SISO) decoding algorithm
- Contributions in FPGA implementation of the SISO decoder

1. Computing the Super Gammas to handle parallel trellis edges
2. An efficient implementation of the maxstar Look-Up-Table
3. Non-blocking interleaver and deinterleaver design.
4. Fast clipping circuit for quantization.

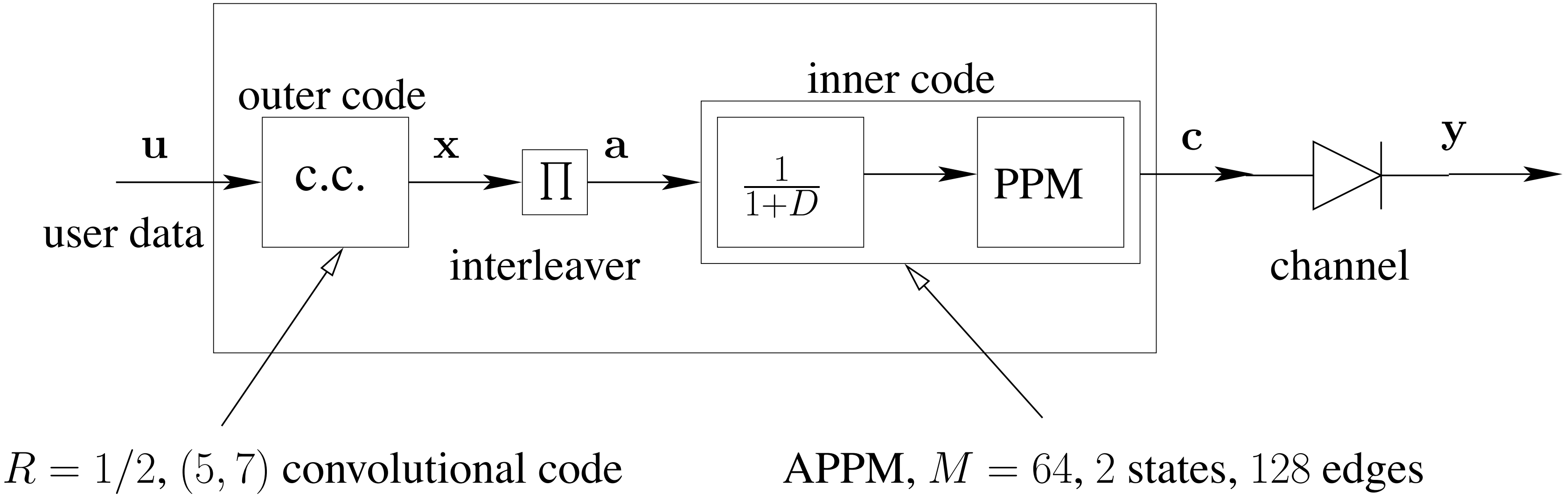
- Hardware speed and throughput

An optical communication system



- Our focus is only on the encoder and decoder.

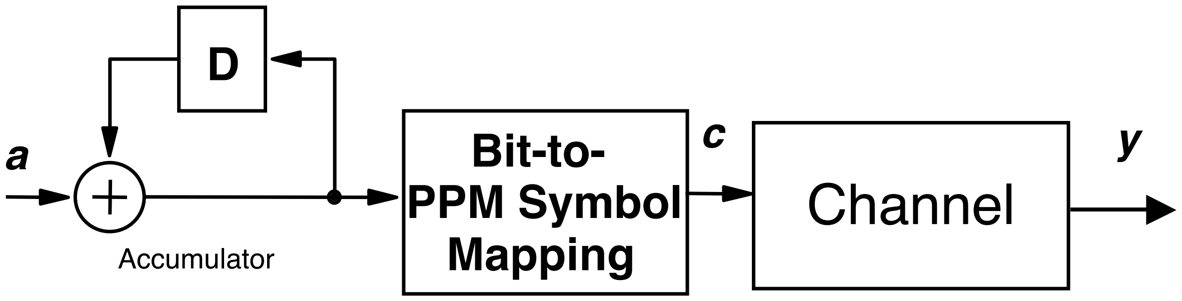
SCPPM Encoder

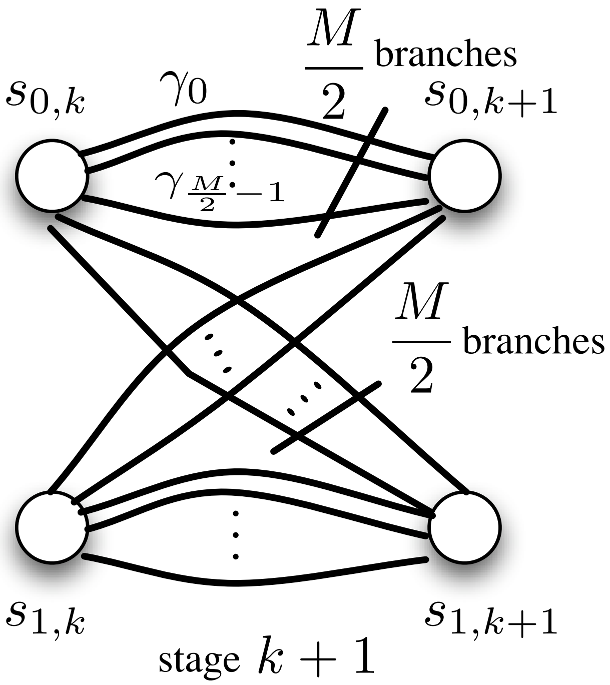


- Serial concatenated pulse-position modulation (SCPPM) encoder
- An outer (5,7) rate 1/2 convolutional code
- An inner accumulate and PPM mapping.
- Possible PPM orders $M = 32, 64, or 128$

The Inner Code

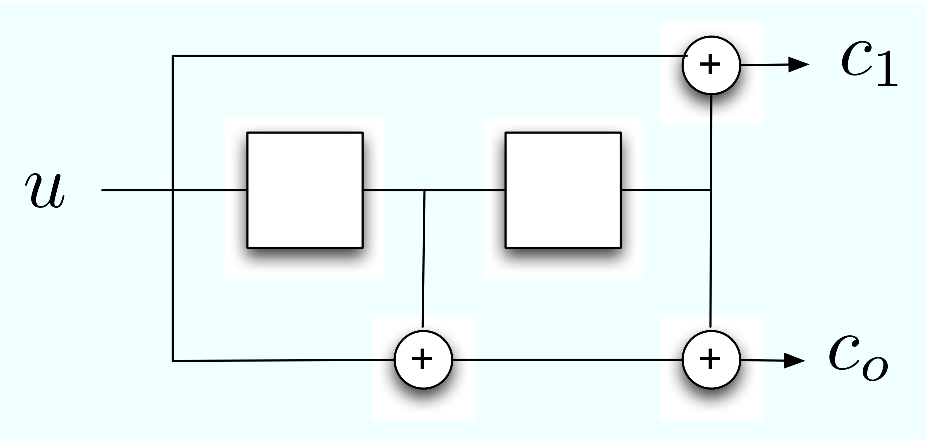
- Accumulator with anti-Gray bit-to-symbol mapping.
- The inner code is described by a 2-state trellis with M/2 edges between connecting states.

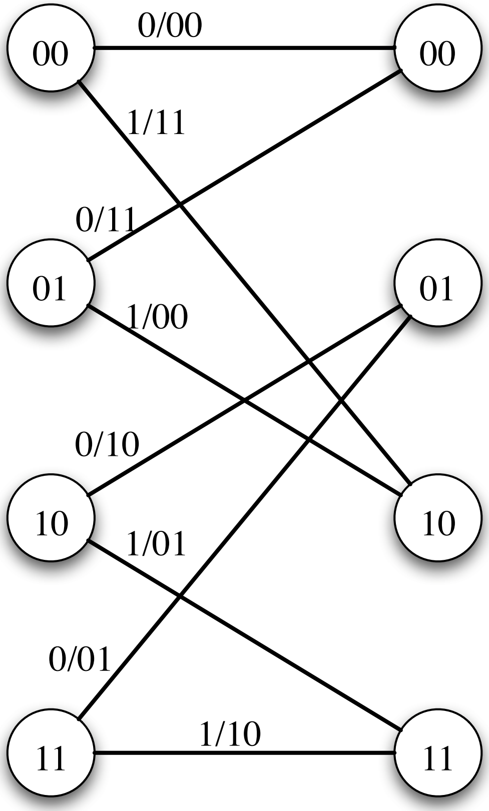




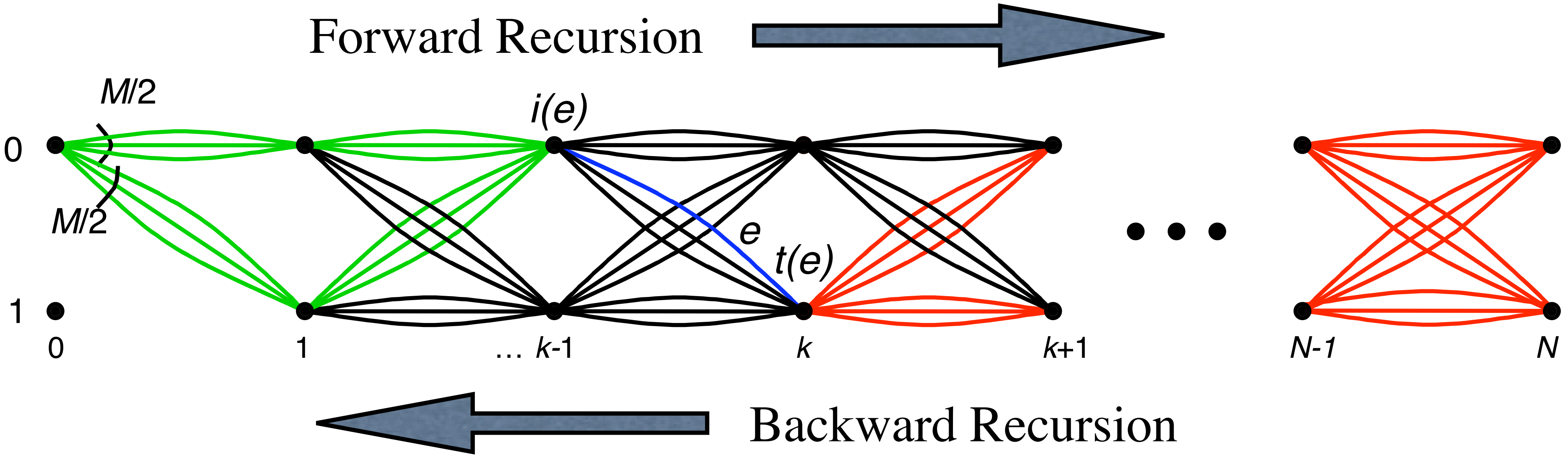
The Outer Code

- The (5,7) convolutional code with rate 1/2.
- The code is described by a 4-state trellis with input/output: $u/c_1, c_0$





Soft-In Soft-Out (SISO) Decoding

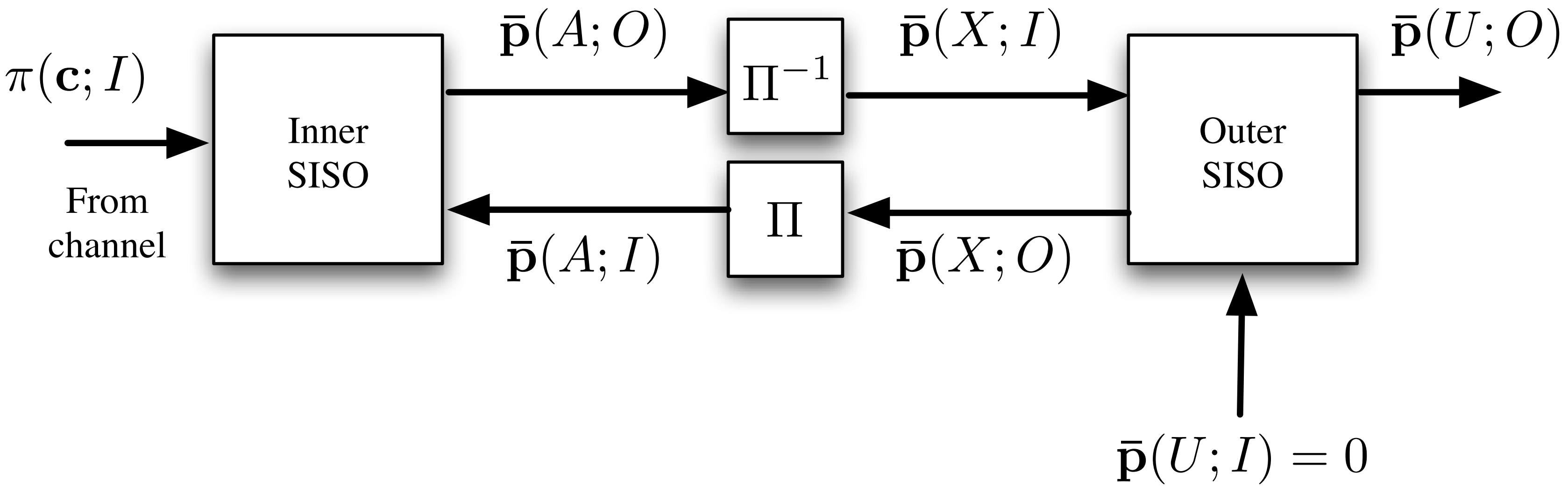


Forward Recursion

Backward Recursion

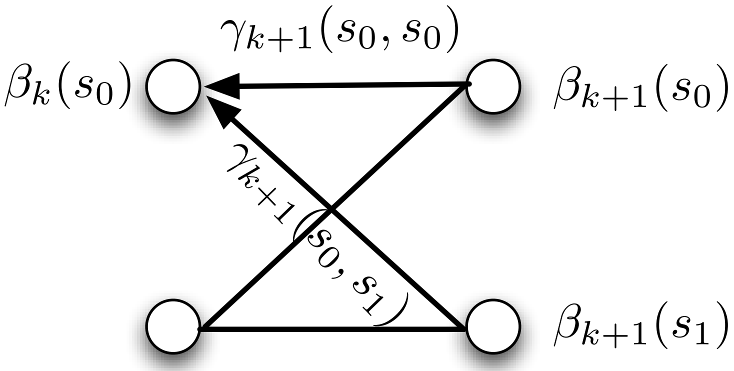
- The BCJR algorithm is used to traverse the trellis and calculate:
- The joint probability that the state is *i(e)* at time k-1 and the sequence y_1^{k-1} is observed -- $\alpha_{k-1}(i(e))$
- The conditional probability that the sequence $y_k^n$is observed starting at time k given current state is *t(e)* -- $\beta_k(t(e))$
- The conditional probability that the transition y is observed given the state pairs i(e) and t(e) -- $\gamma_k(e)$
- The probability of edge e at time k given observation at all times -- $\lambda_k(e)$

Decoder Architecture



- Apply the BCJR algorithm to the inner and outer trellis.
- The bit Log Likelihood Ratios (LLRs) are calculated from the metric $\lambda's$
- Iterate between the inner and outer decoder to refine the overall bit-decision.
- Use the log-domain approach to avoid multiplication and division.

The Maxstar Operation



$$\beta_k(s_0) = \beta_{k+1}(s_0)\gamma_{k+1}(s_0, s_0) + \beta_{k+1}(s_1)\gamma_{k+1}(s_0, s_1)$$

$$\bar{\beta}_k(s_0) = \log(\beta_k(s_0))$$

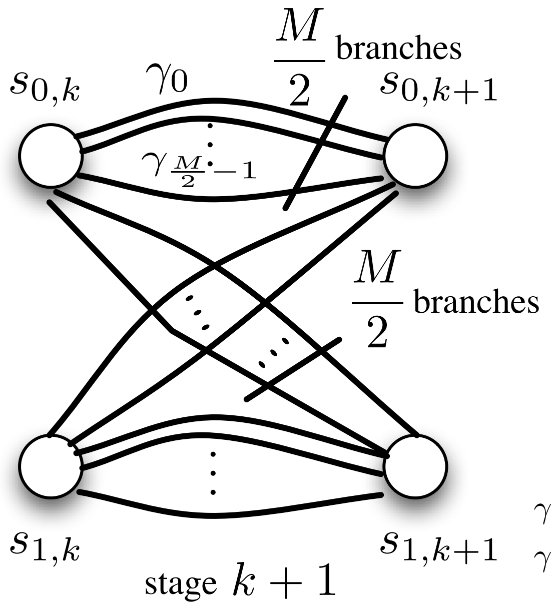
$$\begin{aligned}\bar{\beta}_k(s_0) = \log(&\exp(\bar{\beta}_{k+1}(s_0) + \bar{\gamma}_{k+1}(s_0, s_0)) \\ &+ \exp(\bar{\beta}_{k+1}(s_1) + \bar{\gamma}_{k+1}(s_0, s_1)))\end{aligned}$$

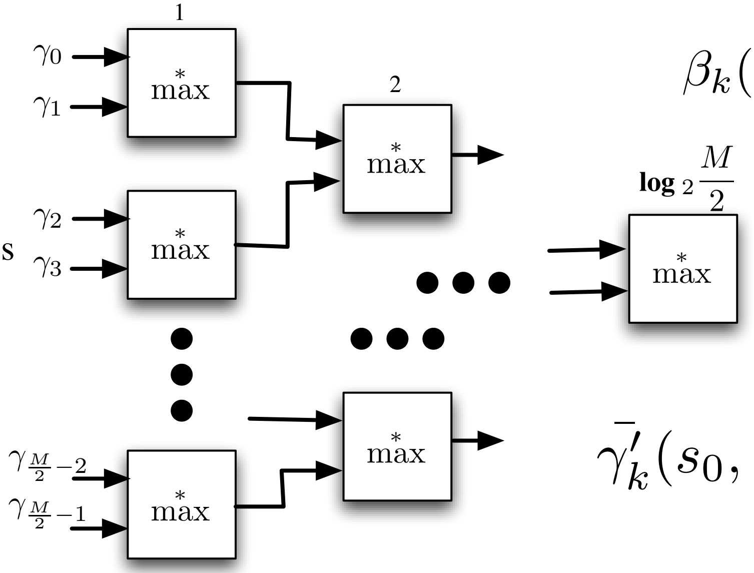
$$\boxed{= \overset{*}{\max}(\bar{\beta}_{k+1}(s_0) + \bar{\gamma}_{k+1}(s_0, s_0), \bar{\beta}_{k+1}(s_1) + \bar{\gamma}_{k+1}(s_0, s_1))}$$

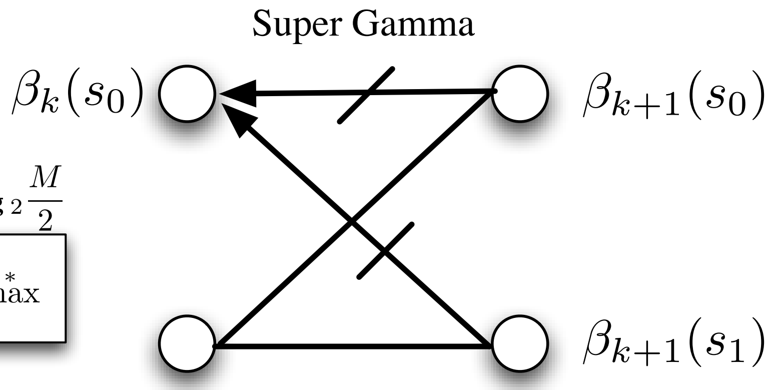
$$\overset{*}{\max}(x, y) = \max(x, y) + \log(1 + e^{-|x-y|})$$

- Can pre-compute the log term in maxstar and store in a Look-Up-Table (LUT)

Super Gammas







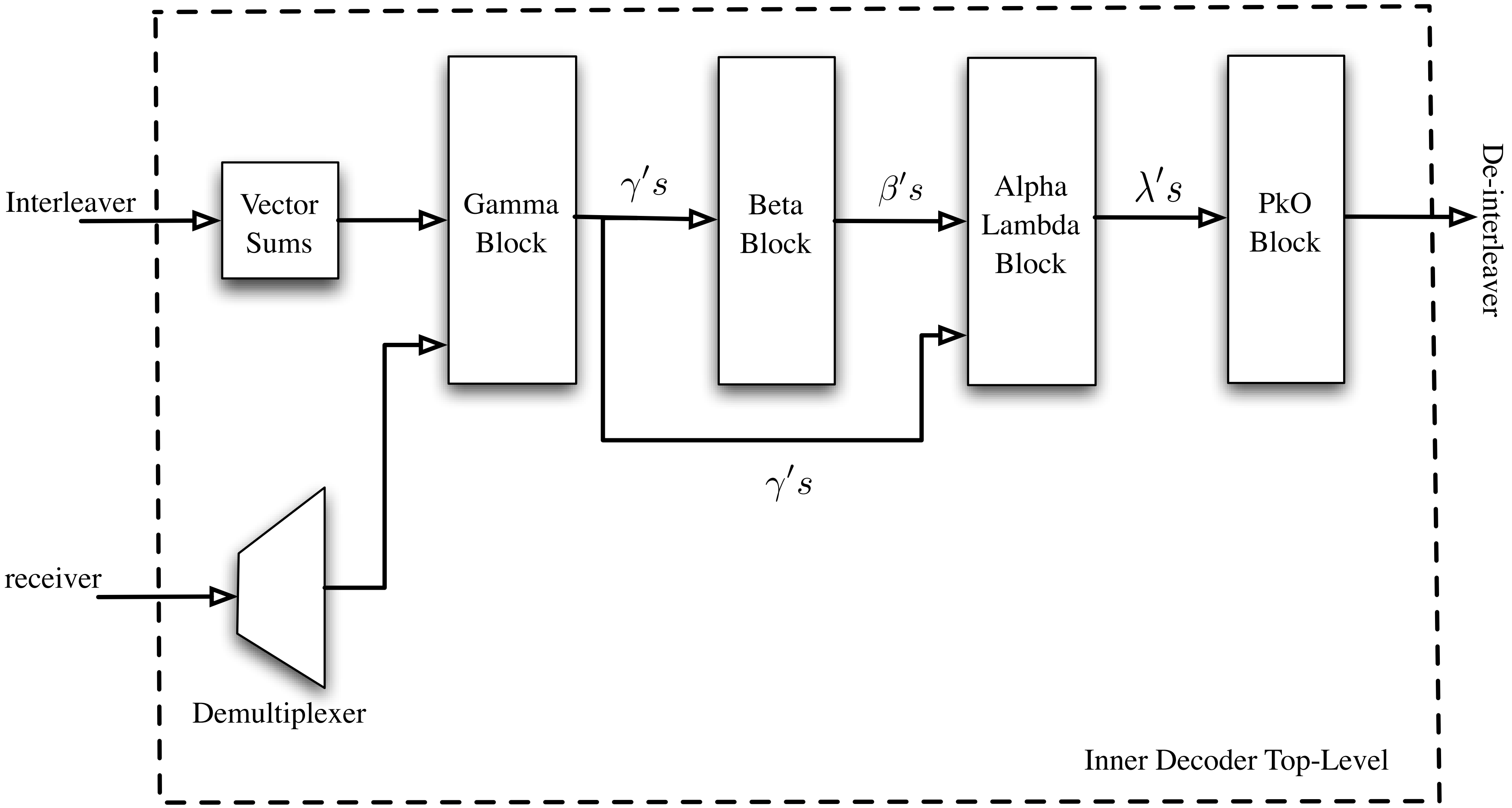
$$\bar{\gamma}'_k(s_0, s_0) = \overset{*}{\max}(\gamma_0, \gamma_1, \cdots, \gamma_{\frac{M}{2}-1})$$

$$\bar{\beta}(s_0) = \overset{*}{\max}(\bar{\beta}_{k+1}(s_0) + \bar{\gamma}'_{k+1}(s_0, s_0),$$
$$\bar{\beta}_{k+1}(s_1) + \bar{\gamma}'_{k+1}(s_0, s_1))$$

But what happens if we have parallel edges?

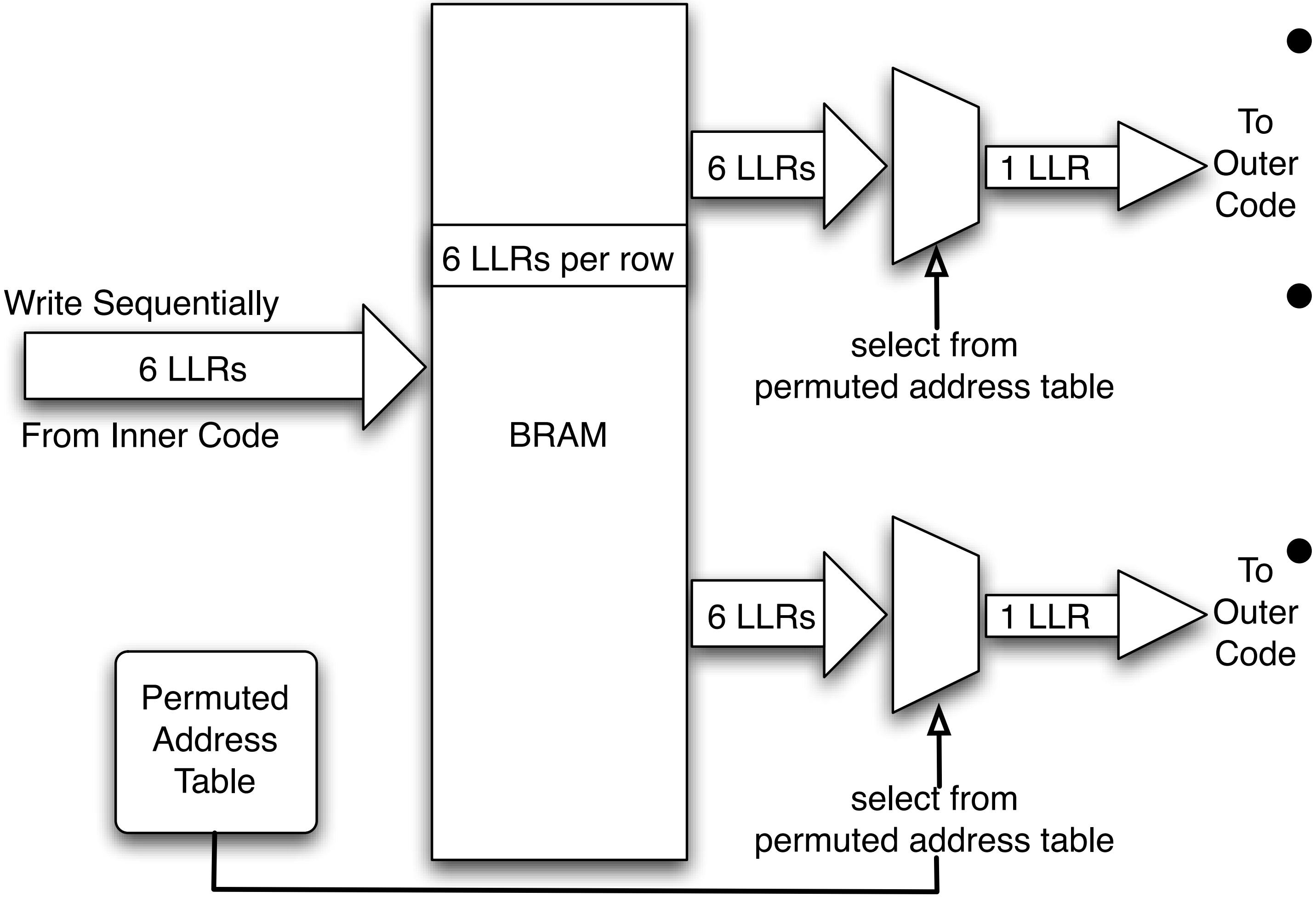
- We can group the parallel gammas into a super gamma and use a pipelined maxstar to compute the super gammas in one clock cycle for the forward and backward recursions.

Block Diagram of Inner Decoder



- Use of partial statistics to reduce data transfer from receiver.
- The outer decoder consists of similar data flow.

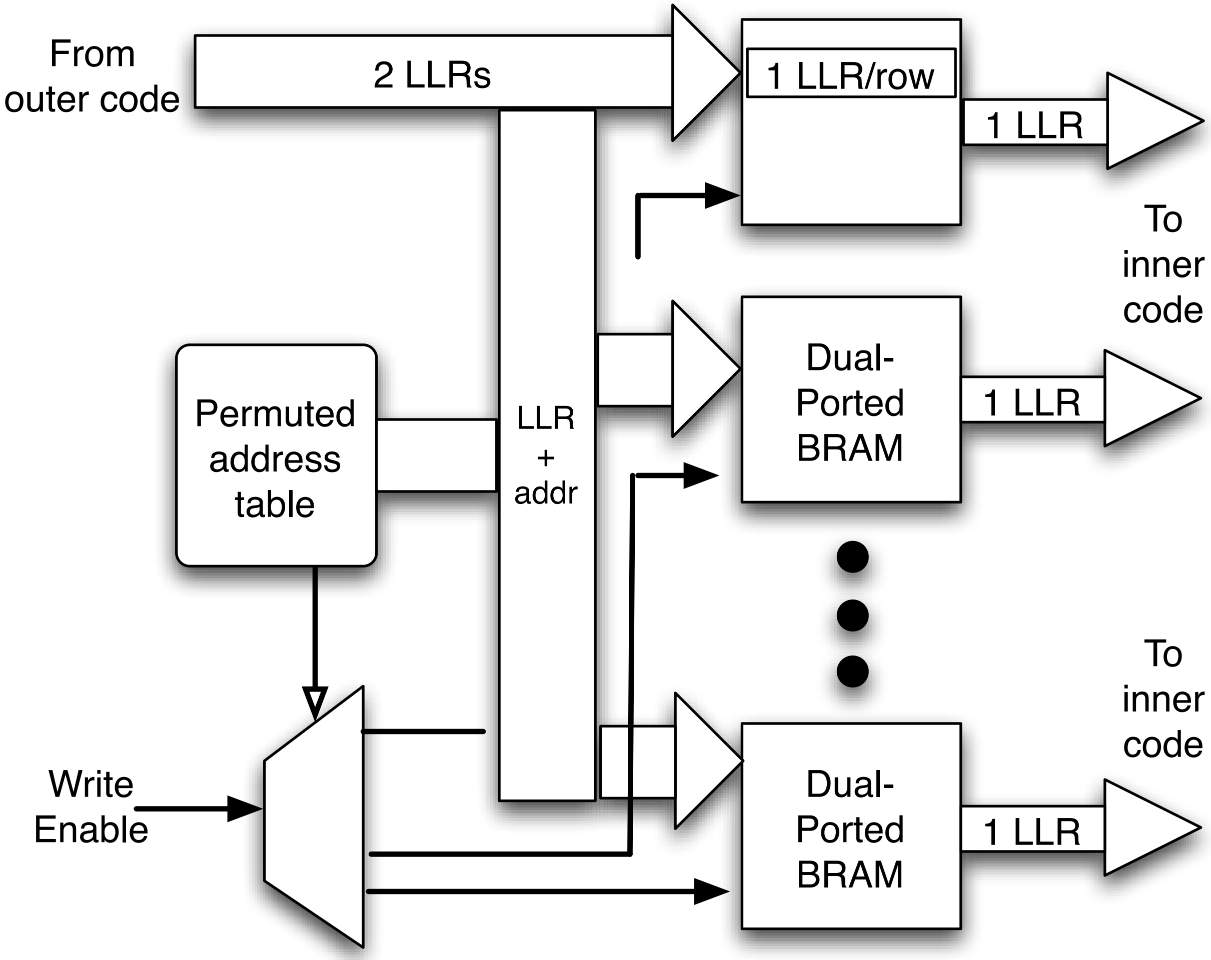
De-interleaver Design



No read conflicts will occur using our mapping.

- Inner code generates 6 LLRs per clock cycle.
- The LLRs are written sequentially into the de-interleaver.
- The outer code fetches 2 LLRs per clock. This read is done by finding the 2 distinct rows that contain the LLRs and selecting the right one out of the 6.

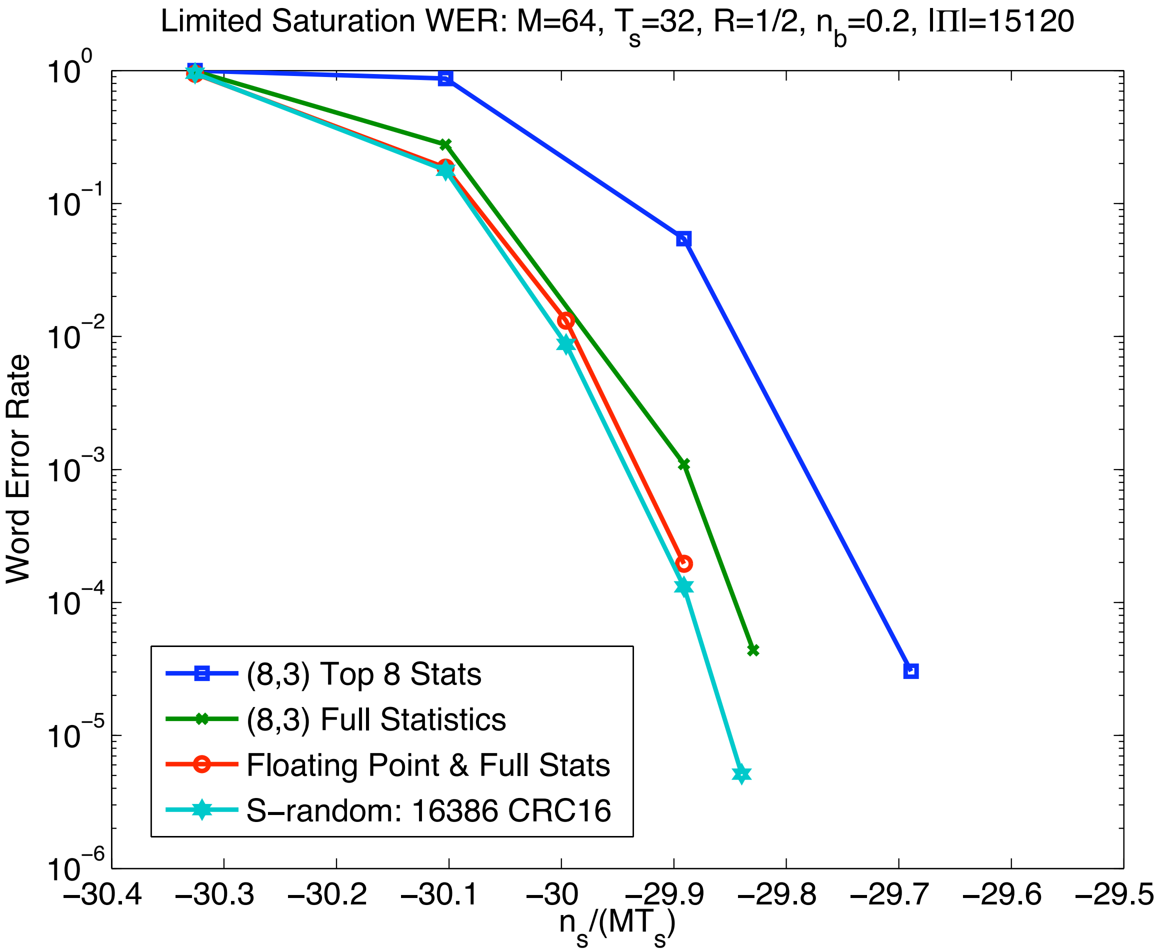
Interleaver Design



- Partition Interleaver into 6 distinct memories
- Write permuted the 2 LLRs generated by the outer code in one clock into the BRAMs
- Our mapping prevents write conflicts
- Even if there were, the dual-ported BRAM can handle simultaneous writes

FPGA Implementation - Quantization Effects

- FPGA implementation requires all floating-point values to be quantized stored as fix-point numbers:
  $q \in [-2^{w-p-1}+1, 2^{w-p-1}-1]$
- w determines dynamic range and p determines decimal precision
- Simulations indicate $w=5$ and $p=3$ would lead to a signal energy loss less than 0.2 dB from floating point performance.



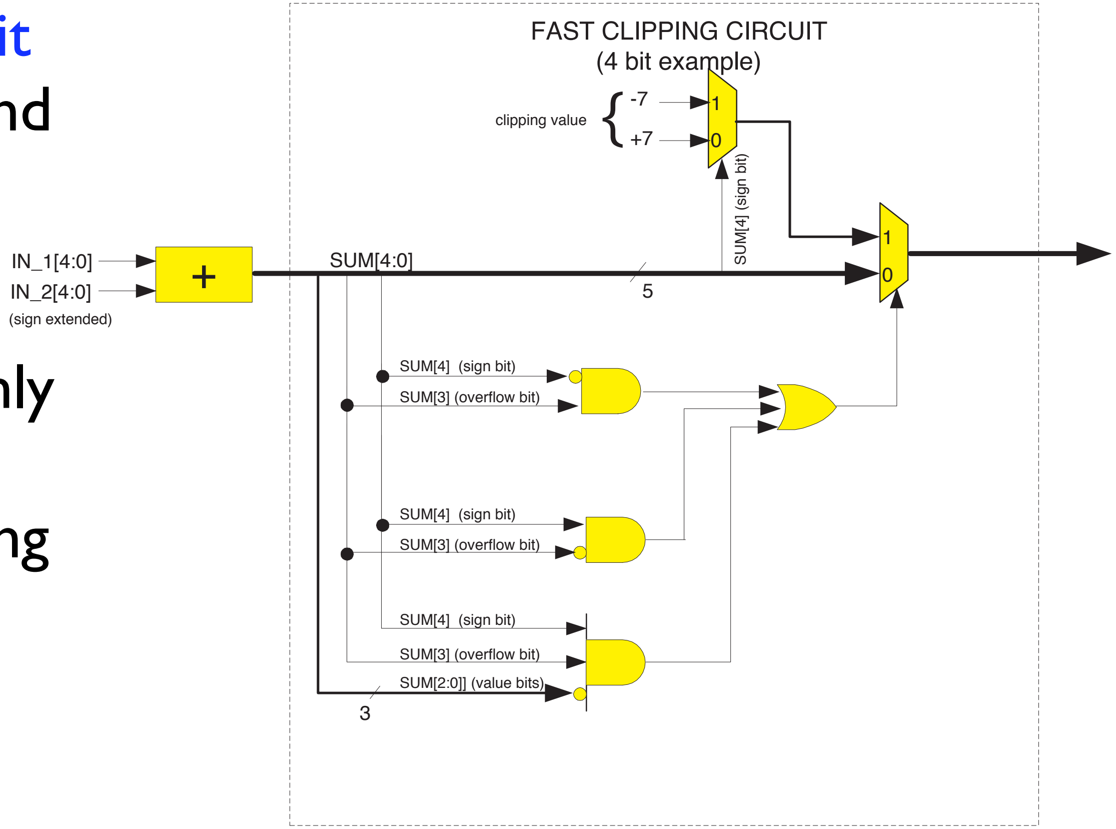
Information Processing Group

FPGA - Memories and Circuits

- We designed a fast clipping circuit to quantize only the log alphas and betas. The other metrics are allowed to grow.

- The maxstar LUT for p=3 has only 21 3-bit entries and this allowed for implementation as ROMs using Xilinx's distributed RAMs.

- Other storage elements such as ones for the betas and channel LLRs are implemented using Xilinx Block RAMs (BRAMs).



Hardware Specifications

- The SCPPM decoder for M=64 is implemented on a Xilinx Virtex II 8000 grade 4 which sits on a Nallatech BenDATA WS board.
- Xilinx place and route reported design has 6.5 million gates.

Full Decoder	used/total	utilization	Inner Decoder	Outer Decoder	Interleavers & others
BRAM	101/168	60 %	19 % of total	9 % of total	32 % of total
Flip Flops	17311/ 93184	18 %	16 %	1 %	1 %
Slices	30174/ 46592	64%	52 %	6 %	6 %

Speed and Throughput

- Assuming 7 average iterations, we can obtain the clock speed and calculate the throughput for various FPGA parts.

Part Number	Maximum Clock	Data Rate
Virtex II 8000 - grade 4	17 MHz	0.907 Mbps
Virtex II 8000 - grade 5	23 MHz	1.23 Mbps
Virtex II - Pro	28 MHz	1.5 Mbps

Summary

- Presented an FPGA implementation of an SCPPM decoder for Deep Space Laser communication.
- Contributions in Super Gammas pipelining, reducing quantization effects, non-blocking interleaver/de-interleaver designs, and a fast clipping circuit are made.
- Potential decoder improvements can be made:
  1. Applying window-based BCJR to inner and outer decoder for a possible 2 and 4 times the speed up respectively.
  2. Building on a larger FPGA part such as the Virtex IV which is already available on the market and could double the speed.
- Using a baseline of 1.5 Mbps, the final design could in principle deliver 24 Mbps and this translates into a 50 Mbps SCPPM decoder that can fit onto 3 FPGAs.